

УДК 621.3; 681.3

Турченко В.О.Інститут комп'ютерних інформаційних технологій,
Тернопільська академія народного господарства**ЗАСТОСУВАННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ ПІДВИЩЕННЯ
ДОСТОВІРНОСТІ ДАНИХ**

© Турченко В.О., 2000

Запропоновано підвищувати достовірність даних, отримуваних від сенсорів, прогнозуванням їх дрейфу нейронною мережею. Наведено узагальнений алгоритм навчання нейронної мережі на основі трьох груп даних про дрейф сенсорів. Аналізуються експериментальні результати прогнозування дрейфу.

Вступ. Задача підвищення достовірності даних, зокрема від первинних перетворювачів (сенсорів), є однією з найбільш важливих в системах збирання та обробки інформації. При цьому найбільшу нестабільність вносять сенсори і вхідні перетворювачі сприйняття інформації [1]. Застосування одного із найбільш поширених структурно-алгоритмічних методів підвищення точності [2] – методу калібрування – вимагає переривання взаємодії з об'єктом, що обмежує використання методу. Метод прогнозування вільний від цього недоліку, однак його застосування вимагає значних попередніх досліджень. Індивідуальні залежності нестабільності сенсора від впливаючих величин описуються переважно складними математичними функціями. Крім того, індивідуальний характер прогнозу викликає проблемні питання, пов'язані зі значними випадковими складовими. В таких випадках доцільним є прогнозування нестабільності сенсорів за допомогою апарату штучних нейронних мереж (НМ). Це особливо актуально для дистрибутивних сенсорних мереж, в яких використовуються різноманітні сенсори в різних умовах експлуатації. Застосування нейронних мереж для прогнозування дрейфу сенсорів в дистрибутивній сенсорній мережі розглянуто нижче.

Використання нейронних мереж в дистрибутивній сенсорній мережі. Інтелектуальна дистрибутивна сенсорна мережа (ІДСМ) складається з трьох рівнів обробки інформації – сенсорних модулів нижнього рівня, інтелектуальних вузлів обробки даних та комп'ютера-сервера верхнього рівня [4, 5]. Сенсорні модулі перетворюють сигнал сенсора в код, але не мають мережевих властивостей. Тому доцільно використати розміщені поблизу сенсорних модулів інтелектуальні вузли, які, по-перше, реалізують протоколи обміну даними з сенсорними модулями з одного боку і мережею верхнього рівня з іншого; по-друге, реалізують всі виконавчі операції інтелектуальних функцій ІДСМ відносно процесу сприйняття інформації [6].

Центральний комп'ютер (сервер) підтримує функціонування всіх компонентів системи, а головне – процедури самоадаптації та самонавчання в інтересах інтелектуальних вузлів. Його програмне забезпечення складається з програми-супервізора та спеціалізованих програм користувача. Основні функції програми-супервізора – ініціалізація мережі, підтримка під'єднаних інтелектуальних вузлів, налаштування каналів сприйняття інформації і забезпечення заданої точності сприйняття по всіх каналах [5]. Останнє базується на прогнозуванні нестабільності компонентів каналу сприйняття за допомогою НМ. Ефективність прогнозування визначається якістю навчання НМ і похибкою обчислення прогнозованих значень. Аналіз

процесів навчання та прогнозування за допомогою НМ показав, що вони неспівмірні за вимогами до обчислювальних ресурсів. Час навчання на комп'ютерах Pentium-200 становить від десятків секунд до десятків хвилин, а прогнозування – десятки мікросекунд. Навчання за алгоритмом зворотнього розповсюдження помилки вимагає достатнього об'єму пам'яті та відповідного процесора. Тому фази навчання та прогнозування доцільно розділити: навчання виконувати на сервері, а прогнозування – в інтелектуальному вузлі. Такий розподіл обчислювальних потужностей зменшує потоки інформації в мережі і підвищує її живучість. Алгоритм навчання нейронної мережі на сервері ІДСМ пропонується нижче.

Алгоритм навчання нейронної мережі. Аналіз показав, що для штучного збільшення кількості точок навчання НМ можуть використовуватись такі три групи даних:

- *Реальні дані* формуються під час роботи сенсора в складі ІДСМ в реальних умовах використання;
- *Історичні дані* являють собою реальну нестабільність інших сенсорів даного типу за попередні проміжки часу. В кращому випадку це нестабільність сенсорів в аналогічних умовах використання, в гіршому – в близьких умовах використання;
- *Гіпотетичні дані* подаються як математичні моделі узагальнених дрейфів сенсорів цього чи аналогічного типів (наприклад, сенсорів, що відрізняються за конструкцією або номінальними параметрами). Вони формуються за літературними джерелами, результатами науково-технічних досліджень, інформацією виробника тощо. Для забезпечення достатньо високої довірчої ймовірності допустимої похибки ІДСМ гіпотетичні дані повинні відображати гірші варіанти поведінки функцій дрейфу.

Узагальнений алгоритм, що показує порядок початкового використання цих трьох груп даних, наведений на рис.1. Спочатку аналізується тип наявних даних за ієрархічним принципом, і, залежно від результату, виконуються такі процедури:

- *Об'єднання даних* (при відсутності реальних даних) – об'єднання історичних даних, що зберігаються в архіві, з врахуванням умов експлуатації сенсорів;
- *Вибір гіпотетичної матмоделі* (при відсутності історичних даних) із сформованої раніше бази гіпотетичних даних про можливий дрейф сенсорів;
- *Конфігурування НМ* полягає в виборі з бази знань моделі нейронної мережі (кількості нейронів, типу їх функції активації, порогу, вагових коефіцієнтів синапсів нейронів, кількості рівнів НМ та зв'язків між нейронами) і алгоритму навчання [7], які (згідно з результатами аналізу гіпотетичних даних) повинні оптимально відповідати майбутньому характеру функції дрейфу конкретного сенсора;
- *Передавання навченої НМ* на нижній рівень передбачає формування відповідного двійкового файлу і реконфігурацію програмного забезпечення інтелектуального вузла [6].

Очевидно, що використання реальних даних забезпечує найвищу точність прогнозу дрейфу. Однак їх кількість, особливо на початку експлуатації сенсора, недостатня для якісного навчання НМ. Для заміни реальних даних можна використати історичні, але тоді точність прогнозу обмежена випадковою складовою функції дрейфу сенсора і адекватністю умов одержання історичних даних до умов експлуатації сенсора, дрейф якого прогнозується. Тому необхідне поступове витіснення історичних даних реальними, в чому і полягає сенс самонавчання ІДСМ. Точність прогнозу при використанні гіпотетичних даних ще нижча, тому вони повинні, по можливості, використовуватися на початковому етапі експлуатації сенсора.

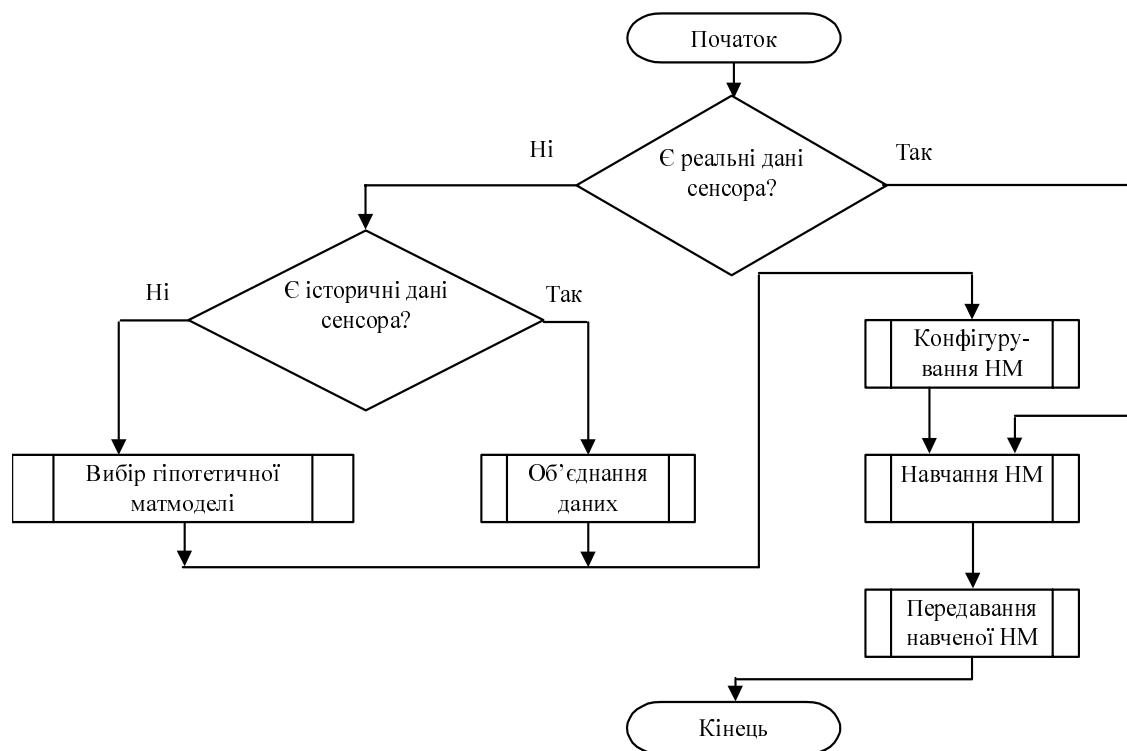


Рис. 1. Узагальнений алгоритм використання трьох груп для навчання нейронної мережі

Експериментальні результати. Використання історичних даних дозволяє зробити надійніші висновки про ефективність прогнозування, однак обмежує їх конкретними випадками наявних даних. Ці дані не дозволяють дослідити можливості самонавчання і адаптації ІДСМ, зробити висновки про її технічні характеристики та залежності цих характеристик від поведінки функції дрейфу сенсорів. Тому для дослідження ІДСМ доцільно використовувати гіпотетичні дані, які повинні відображати, по-можливості, повну множину варіантів функцій дрейфу сенсорів різного типу.

При проведенні експериментальних досліджень гіпотетична нестабільність сенсорів моделювалася як сума повільного дрейфу та різного виду зовнішніх та внутрішніх завад. Повільний дрейф сенсора моделювався функцією квадратного кореня. Детерміновані внутрішні та зовнішні завади моделювалися за допомогою суми функцій синуса різної амплітуди і частоти. Випадкові завади та випадкова складова дрейфу сенсора моделювалися за допомогою генератора випадкових чисел з рівномірним (на інтервалі $\{0;1\}$) і нормальним (математичне очікування 0 та дисперсія 1) законами розподілу.

Прогнозування дрейфу сенсорів здійснювалось в системі MATLAB Neural Network Toolbox. Для прогнозування використана модель трирівневого персептрону, який складався з 6 вхідних, 10 схованих нейронів з функцією активації гіперболічного тангенсу та одного вихідного лінійного нейрону. НМ навчалася на інтервалі $\{0;10\}$, крок навчальної вибірки 0.1. Тривалість навчання не перевищувала 180 епох (4 хвилини на комп'ютері Pentium-200). Результати навчання чотирьох досліджуваних гіпотетичних функцій наведені на рис.2 – 5. Прогнозовані функції подано під кожним рисунком, випадкові складові розподілені для функцій рис. 2-4 згідно з рівномірним законом, а для функції рис.5 – за нормальним законом. Для функцій рис. 2,3 навчання НМ здійснювалось до середньоквадратичної

помилки 10^{-2} , для функцій 4, 5 – до 10^{-3} . Як видно з графіків, реальна та прогнозована криві практично накладаються, відносна похибка прогнозування не перевищує 2%.

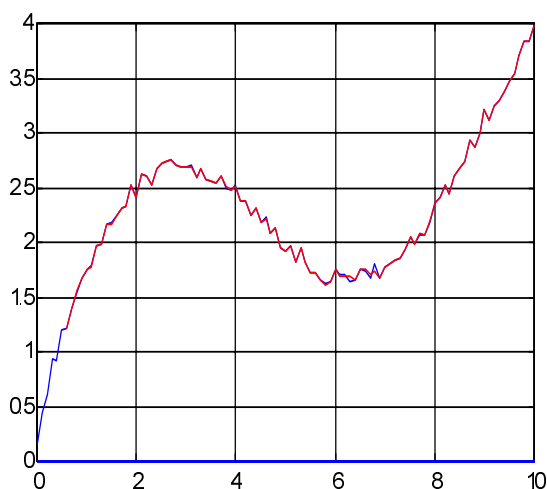


Рис. 2. Прогноз функції $y = \sqrt{x} + \sin(0.7x) + 0.2rand$

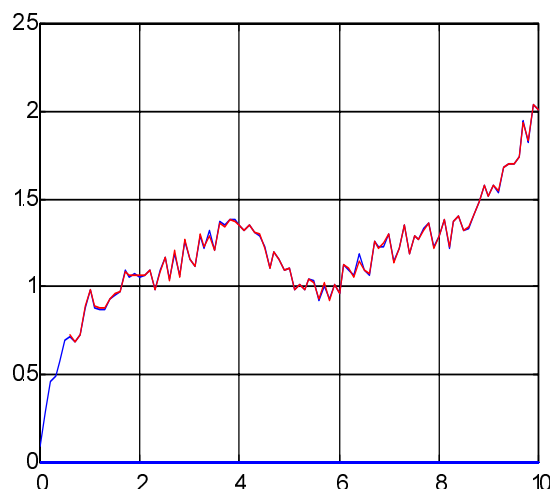


Рис. 3. Прогноз функції $y = 0.7\sqrt{x} + 0.5\sin(0.7x) + 0.2rand$

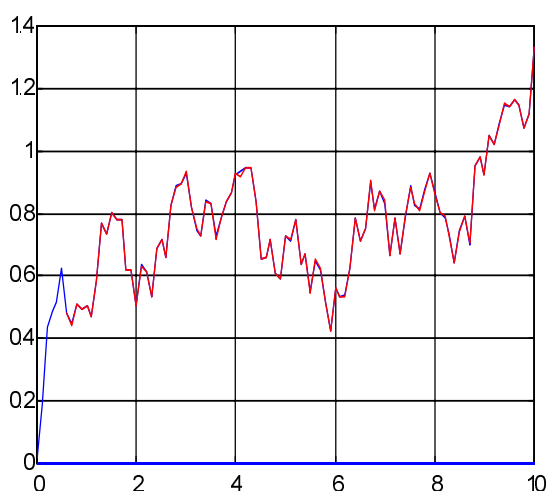


Рис. 4. Прогноз функції
 $y = 0.5\sqrt{x} + 0.3\sin(0.7x) + 0.1\sin(2x) + 0.2rand$

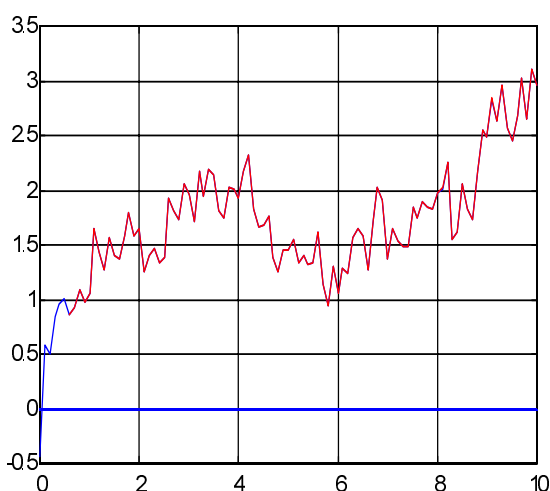


Рис. 5. Прогноз функції
 $y = 0.8\sqrt{x} + 0.6\sin(0.7x) + 0.2\sin(2x) + 0.1\sin(5x) + 0.2rand$

Висновки. Проведені на групі гіпотетичних даних дослідження і отримані результати дозволяють зробити висновок про можливість досягнення високої достовірності даних за рахунок прогнозування нейронними мережами складних функцій дрейфу сенсорів зі значними випадковими складовими.

1. Sachenko A., *Precision Sensory Data Acquisition and Processing for Intelligent Robotic System, Proceedings "EURISCON'94", Malaga (Spain), 1994, vol.3, pp. 1681-1691.* 2. Туз Ю.М. Структурные методы повышения точности измерительных устройств. –К., 1976. 3. Рольберг И.Л., Нужнов А.Г., Покровская Г.Н. и др. Стабильность термоэлектродвижущей силы термопар хромель-алюмель при нагреве на воздухе при температурах до 1200 °С // Иссле-

довані сплавів для термонар: Сб. тр. / Гипроцветметобработка. – М., 1967. – Вып. 24. Т. II. – с. 54-65. 4. Саченко А.О., Кочан В.В., Турченко В.О. Дистрибутивна сенсорна жа підвищеної ефективності / Вісн. ДУ “Львівська політехніка”, 1998. – с77-80. 5. Sachenko A., Kochan V., Turchenko V., *Intelligent Distributed Sensor Network, Proceedings of the 15th IEEE Instrumentation and Measurement Technology Conference, St. Paul, USA – May 18-20, 1998, vol. 1, pp. 60-66.* 6. Sachenko A., Kochan V., Turchenko V., Tymchyshyn V., Vasylykiv N., *Intelligent Nodes for Distributed Sensor Network, Proceedings of the 16th IEEE Instrumentation and Measurement Technology Conference, Venice, Italy - May 24-26, 1999, vol. 3, pp. 1479-1484.* 7. Golovko V., Grandinetti L., Kochan V., Laopoulos T., Sachenko A., Turchenko V. *Sensor Signal Processing Using Neural Networks, Africon'99, Cape Town, South Africa, Sep 29-Oct 1, 1999, in press.*

УДК 681.3

Черкаський М.В.

ДУ “Львівська політехніка”, кафедра ЕОМ

МОДЕЛІ НЕФОРМАЛЬНИХ АЛГОРИТМІВ

© Черкаський М.В., 2000

Розглядаються дві моделі неформальних алгоритмів - алфавітний оператор і блок-схема (система Ляпунова). Вказуються способи зменшення часової складності алфавітних операторів та структурної складності блок-схеми алгоритмів.

Вступ. Словесне визначення поняття алгоритму як точного припису, який задає обчислювальний процес, що починається з довільного початкового даного (з деякої сукупності можливих для даного алгоритму початкових даних) і спрямований на отримання повністю визначеного цим початковим даним результату [1] лише в неявній формі перераховує сім параметрів алгоритму: систему початкових даних і умов, системи проміжних і кінцевих результатів, правило початку і закінчення, правило безпосереднього перероблення і правило виводу результатів, але не вказує, як утворюються ці параметри для конкретних задач.

Словесне тлумачення алгоритму є недостатнім для дослідження ряду математичних проблем, до яких зокрема належала необхідність доведення того, що деяка задача не має розв'язання. Тому виникла потреба в моделях, які дозволяли б обмеженою кількістю загально зрозумілих елементарних операцій розв'язувати будь-яку задачу або доводити, що задача не має розв'язання. Було запропоновано декілька уявних (математичних) моделей, які одержали назву формальних алгоритмічних систем (ФАС). ФАС ініціювали розвиток матричної теорії алгоритмів, предметом досліджень якої були характеристики складності алгоритмів. Прикладами ФАС є машина Тьюрінга-Поста, неформальні алгоритми Маркова, рекурсивні функції тощо. З точки зору наближення математичних моделей до реальних обчислювальних машин перевагу серед ФАС має машина Тьюрінга.

Машина Тьюрінга дозволяє визначати декілька характеристик складності алгоритмів розв'язання задач – часову, місткісну, опису програми. Часова складність оцінюється кількістю елементарних кроків машини Тьюрінга, яка потрібна для розв'язання задачі. Місткісна складність оцінюється кількістю комірок пам'яті, яка потрібна для розв'язання задачі. Складність опису програми – це кількість команд програми.